Implementing Automated Security Monitoring using OpenSCAP

Graham Mosley – University of Pennsylvania
Mentors: George Rumney, John Jasen, NASA GSFC Code 606.2 (NCCS)

NASA Center for Climate Simulation

Abstract

In order for NASA High Performance Computing systems to be operational, they must be secure. That security is greatly enhanced using continuous monitoring of Configuration and Control compliance with standardized tools. This internship focused on research of automated security monitoring for use with the NASA Center for Climate Simulation (NCCS). The National Institute of Standards and Technology provides a protocol for automated security testing called SCAP (Security Content Automation Protocol). Since SCAP is just a protocol, a program is needed that implements it. For this project, OpenSCAP, a free open-source implementation maintained by Red Hat was chosen. The OpenSCAP tool was used to evaluate vulnerabilities and configurations of CentOS, SUSE Linux Enterprise Server, and Debian Linux distributions. In addition to testing the OpenSCAP tool, Perl scripts were written to improve automation and tailor it to the needs of the NCCS by integrating it with NAGIOS monitoring using passive service checks. Currently, this project is still in the testing phase. However, early results for RedHat, CentOS and SUSE Linux Enterprise Sever are promising and may be implemented into the NCCS operational environment.

What is SCAP?

The Security Content Automation Protocol (SCAP) is a set of standards for security monitoring created by NIST. There are many components that make up SCAP. The main components tested were:

- Extensible Configuration Checklist Description Format (**XCCDF**) – A set of criteria usually in the form of a benchmark that makes sure a system is configured correctly to a set of specifications.
- Common Vulnerabilities and Exposures (**CVE**) – A format for publicly known security vulnerabilities.
- Common Vulnerability Scoring System (**CVSS**) – A method for scoring the severity of a CVE based on a criteria.
- Open Vulnerability Scoring System (**OVAL**) – a language used to test and evaluate specific characteristics of a machine. OVAL is normally used to evaluate XCCDFs and to check for CVEs.

Since SCAP is just a protocol, we need a program that implements it. For this project, OpenSCAP – a free open source implementation was chosen. While the package is offered in many default repositories, it was decided to build a newer version of it.

SUSE CentOS debian

The main distributions used for testing were SLES11 SP3, CentOS 7/8 and Debian 8

OpenSCAP

OpenSCAP was chosen to evaluate SCAP content

Obtaining SCAP Content

This project collected and tested two different types of SCAP content, security benchmark checklists and OVAL CVE streams.

Security Benchmarks

SCAP SECURITY GUIDE — Center for Internet Security®

Benchmarks are a set of rules that a system must adhere to. One rule could be that the minimum password length requirement must be at least 12 characters. Each rule can have a severity level ranging from low to high. A scan will run all tests and determine a compliance score from 0 to 100.

The SCAP Security Guide is an open source project that provides security benchmark configuration files for Red Hat Enterprise Linux and its derivatives. Additionally, NASA provides a benchmark for CentOS 6 that is derived from a SCAP Security Guide benchmark.

The Center for Internet Security (CIS) provides benchmarks for CentOS 7 and SLES 11 SP3 among others. These benchmarks are extensive and contain over 100 tests. Unfortunately CIS includes extra content in their security benchmarks that only work with their proprietary CIS-CAT tool that requires JAVA. The benchmarks were modified to work with OpenSCAP.

OVAL CVE feeds

CVE-2014-0160

Impact:	Important
Public:	2014-04-07
CWE:	CWE-130->CWE-119->CWE-201
Bugzilla:	1084875: CVE-2014-0160 openssl: information disclosure in handling of TLS heartbeat extension packets
IAVA:	2012-A-0104

Details

An information disclosure flaw was found in the way OpenSSL handled TLS and DTLS Heartbeat Extension packets. A malicious TLS or DTLS client or server could send a specially crafted TLS or DTLS Heartbeat packet to disclose a limited portion of memory per request from a connected client or server. Note that the disclosed portions of memory could potentially include sensitive information such as private keys.

Every security vulnerability has a corresponding CVE. CVEs contain information about a security problem in a certain version of a program. When CVEs are released, a patch for the program is usually released soon after.

Since NCCS servers do not run automatic updates, we need a way to determine if there are any packages with outstanding CVEs and how severe they are. Both Red Hat and SUSE provide constantly updated OVAL tests to check for any packages affected by CVEs.

While SUSE/Novell's feed for SLES works right out the box, the Red Hat feed must be heavily modified for use with CentOS. The tests for operating system must be replaced with the corresponding CentOS version. In addition to changing platform information and signing keys to the CentOS equivalent, extra work is needed for the Red Hat packages modified by CentOS. When CentOS modifies a package from the Red Hat sources, they strip the minor OS version and add "centos" to the package name. Therefore, the tests for CentOS modified packgages must be changed. This process was automated with Perl script that uses XML parsing.

OVAL Results Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	cpe:/a:open-scap:oscap	1.2.3	2015-07-17	10:11:27
#✕	#✓	#Error	#Unknown	#Other
4	2630	0	0	0

OVAL Definition Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	Red Hat OVAL Patch Definition Merger	3	2015-07-17	04:10:01
#Definitions	#Tests	#Objects	#States	#Variables
2636 Total: 0 0 0 2636 0	23213	2344	4053	0

OVAL System Characteristics Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	cpe:/a:open-scap:oscap		2015-07-17	10:11:27

OVAL Definition Results

✕ | ✓ | Error | Unknown | Other

ID	Result	Class	Reference ID	Title
oval:com.redhat.rhsa:def:20151137	true	patch	[RHSA-2015:1137-01], [CVE-2014-9420], [CVE-2014-9529], [CVE-2014-9584], [CVE-2015-1573], [CVE-2015-1593], [CVE-2015-1805], [CVE-2015-2830]	RHSA-2015:1137: kernel security and bug fix update (Important)
oval:com.redhat.rhsa:def:20150987	true	patch	[RHSA-2015:0987-00], [CVE-2015-3331]	RHSA-2015:0987: kernel security and bug fix update (Important)
oval:com.redhat.rhsa:def:20150726	true	patch	[RHSA-2015:0726-00], [CVE-2014-8159], [CVE-2015-1421]	RHSA-2015:0726: kernel security and bug fix update (Important)
oval:com.redhat.rhsa:def:20141801	true	patch	[RHSA-2014:1801-00], [CVE-2014-3675], [CVE-2014-3676], [CVE-2014-3677]	RHSA-2014:1801: shim security update (Moderate)
oval:com.redhat.rhsa:def:20151243	false	patch	[RHSA-2015:1243-00], [CVE-2015-2590], [CVE-2015-2601], [CVE-2015-2621], [CVE-2015-2625], [CVE-2015-2627], [CVE-2015-2628], [CVE-2015-2632], [CVE-2015-2637], [CVE-2015-2638], [CVE-2015-2664], [CVE-2015-2808], [CVE-2015-4000], [CVE-2015-4731], [CVE-2015-4732], [CVE-2015-4733], [CVE-2015-4748], [CVE-2015-4749], [CVE-2015-4760]	RHSA-2015:1243: java-1.6.0-sun security update (Important)

A human-readable report generated by the Centos CVE script

Nagios Integration

Nagios is an open source network monitoring tool. It is used by the NCCS to make sure that all devices on their network are running correctly. Nagios provides support for passive service checks. Each passive check can result in OK, WARN, or CRITICAL along with a short description. A Nagios check was written for both the Security Benchmark and CVE tests. The checks are written in a way that allows for easy customization. Currently, the benchmark test allows for a warning/critical based on the final compliance score and the CVE check based on the amount of vulnerable CVEs. Both checks also report the number of failed tests/CVEs along with their severity.

Next Steps

Currently, the Nagios scripts are running in the NCCS testing network. The final step will be adding the Nagios scripts to production servers. While collecting SCAP material for SUSE and CentOS proved to be quite successful, it was difficult to find anything for Debian. The CIS Benchmark for Debian is very old and Debian stopped providing an OVAL CVE feed in early 2011.

Acknowledgements

I would like to acknowledge my mentors George Rumney and John Jasen for their help and guidance throughout the internship. I would also like to thank the members of the NASA NCCS security group for their input and help, and Max Guillen and Zed Pobre for their help with NCCS Nagios integration.

scap.nist.gov/
www.open-scap.org/
github.com/OpenSCAP/
benchmarks.cisecurity.org/
redhat.com/security/data/metrics/
support.novell.com/security/oval/